Document de présentation – FNOOK

1. Présentation FNOOK

Fnook est un site web « marketplace » qui vend des biens issus du célèbre jeu « animal crossing New Horizons » sur nitendo SWITCH. Tout utilisateur enregistré a le droit de vendre et acheter sur cette plateforme.

* Un vendeur a la possibilité de choisir sur une liste définit le bien qu’il souhaite mettre en vente uniquement s’il le possède évidemment. Il pourra définir son prix et sa quantité.
* Un client a la possibilité de créer un panier. Commander à plusieurs particuliers différents et choisir l’offre la plus intéressante sur un produit. Après validation du panier une facture sera créée.

Quant aux invités ils peuvent voir toutes les offres, ajouter leur préféré au panier sans possibilité de le valider.

1. Rôles

Réalisation de l’API et de la BDD -> Adrien LEIB

Réalisation de l’IHM et de la BDD -> Hugo Jean

1. Technologies utilisées

API :

* NodeJS

IHM :

* Framework : REACT

BDD :

* Mongoose

1. Architecture Logiciel

Service

(React-Fetch)

Model

(Mongoose)

Database

(mongodb)

IHM

(Localhost :3000)

Pages

(React)

Controller

API

(localhost :3030)

Router

(Express JS)

Router

(Reat-Router)

1. Fonctionnalité majeure
2. **Depuis l’API :**

url\_api: <http://localhost:3030/api/v1>

L’ensemble des requêtes nécessitent d’être utilisé un « x-access-token » à ajouter au header que vous pouvez récupérer à l’aide des url d’authentifications.

1. **Utilisateur**

* « url\_api »/client/register
  + POST
  + Retourne clef token
  + Dictionnaire
  + Permet de créer un utilisateur.
  + Droit : administrateur, utilisateur et invité
* « url\_api »/client/login
  + POST
  + Retourne une clef token.
  + Permet à un utilisateur enregistré de se connecter, ainsi il pourra bénéficier de l’ensemble des fonctionnalités du service.
  + Droit : administrateur, utilisateur et invité.
* « url\_api » /client
  + GET
  + Retourne au format JSON l’ensemble des utilisateurs.
  + Liste de dictionnaire.
  + Droit : administrateur, utilisateur et invité.
* « url\_api » /client/:id
  + GET
  + Retourne au format JSON un utilisateur à partir de son identifiant (id)
* « url\_api » /client/:id
  + PATCH
  + Retourne au format JSON l’utilisateur modifié
  + Permet à un administrateur ou l’utilisateur concerné de modifier l’ensemble de ces données
  + Droit : administrateur, utilisateur concerné.
* « url\_api » /clientPassword/:id
  + PATCH
  + Retourne au format JSON l’utilisateur avec son mot de passe modifié
  + Permet à un administrateur ou l’utilisateur concerné de modifier son mot de passe
  + Droit : administrateur, utilisateur concerné.
* « url\_api » /client/:id
  + DELETE
  + Supprime un utilisateur
  + Droit : administrateur.

1. **FACTURE**

* « url\_api » /invoice
  + POST
  + Retourne au format JSON une facture
  + Permet à un client de créer une facture.
  + Droit : administrateur et utilisateur.
* « url\_api » /invoice
  + GET
  + Retourne au format JSON l’ensemble des factures
  + Liste de Dictionnaire
  + Droit : administrateur.
* « url\_api » /invoice/:id
  + GET
  + Retourne au format JSON une facture à partir de son identifiant (id)
  + Dictionnaire
  + Droit : administrateur.
* « url\_api » /invoice/:id
  + PATCH
  + Retourne au format JSON la facture modifiée
  + Permet à un client de modifié une facture/
  + Droit : utilisateur concerné.
* « url\_api » /invoice/:id
  + DELETE
  + Retourne au format JSON la facture supprimée
  + Supprime une facture à partir de son identifiant (id)
  + Droit : administrateur.
* « url\_api » /invoice/delete/all
  + DELETE
  + Retourne toutes les factures supprimées au format JSON.
  + Liste de dictionnaire.
  + Supprime l’ensemble des factures présente sur le site.
  + Droit : administrateur.

1. **PRODUIT MERE**

* « url\_api » /motherproduct
  + POST
  + Retourne à le produit mère crée.
  + Permet à administrateur de créer un produit mère.
  + Dictionnaire.
  + Droit : administrateur.
* « url\_api » /motherproduct
  + GET
  + Retourne au format JSON l’ensemble des produits mères.
  + Dictionnaire.
  + Droit : administrateur, utilisateur et invité.
* « url\_api » /motherproduct/:id
  + GET
  + Retourne au format JSON le produit mère à partir de son identifiant (id)
  + Dictionnaire
  + Droit : tous.
* « url\_api »/motherproduct/:id
  + PATCH
  + Retourne au format JSON le produit mère modifié
  + Permet à un administrateur de modifié un produit mère.
  + Dictionnaire.
  + Droit : administrateur.
* « url\_api » /motherproduct/:id
  + DELETE
  + Retourne au format JSON le produit mère supprimé
  + Permet à un administrateur de modifié un supprimer un produit mère.
  + Dictionnaire
  + Droit : administrateur
* « url\_api » /motherproduct/delete/all
  + DELETE
  + Retourne au format JSON l’ensemble des produits mères supprimé
  + Permet à un administrateur de supprimer l’ensemble des produits mères.
  + Liste de Dictionnaire
  + Droit : administrateur

1. **PRODUIT**

* « url\_api » /product
  + POST
  + Retourne au format JSON le produit crée.
  + Permet à utilisateur connecté de créer un produit.
* « url\_api » /product
  + GET
  + Retourne au format JSON l’ensemble des produits
  + Liste de dictionnaire
  + Disponible à tous
* « url\_api » /search/:name
  + GET
  + Retourne au format JSON tous produits contenant le « name » saisie.
  + Liste de dictionnaire
  + Droit : administrateur, utilisateur et invité
* « url\_api » /productOfMother/:id
  + GET
  + Retourne au format JSON tous produits liés à l’identifiant (id) mère saisie.
  + Liste de dictionnaire
  + Droit : administrateur, utilisateur et invité.
* « « url\_api » /product/:id
  + Get
  + Retourne au format JSON le produit lié à l’identifiant saisie.
  + Dictionnaire
  + Droit : administrateur, utilisateur et invité.
* « url\_api » /product/:id
  + PATCH
  + Retourne au format JSON le produit modifié liée à l’identifiant saisie.
  + Dictionnaire
  + Droit : administrateur, utilisateur (créateur)
* «« url\_api » /product/:id
  + DELETE
  + Retourne au format JSON le produit supprimé liée à l’identifiant (id) saisie.
  + Dictionnaire
  + Droit : administrateur, utilisateur (créateur)
* « url\_api » /product/delete/all
  + DELETE
  + Retourne au format JSON l’ensemble des produits supprimés.
  + Supprime tous les produits.
  + Dictionnaire
  + Droit : administrateur, utilisateur
* « url\_api » /productofseller/:id
  + GET
  + Retourne au format JSON l’ensemble des produits d’un vendeur à partir de son identifiant
  + Liste de dictionnaire
  + Droit : administrateur, utilisateur, invité
* « url\_api » /invoicesofclient/ :id
  + GET
  + Retourne au format JSON l’ensemble des factures liée à un client à partir de son identifiant
  + Liste de dictionnaire
  + Droit : Utilisateur concerné, administrateur.

1. **MCD**

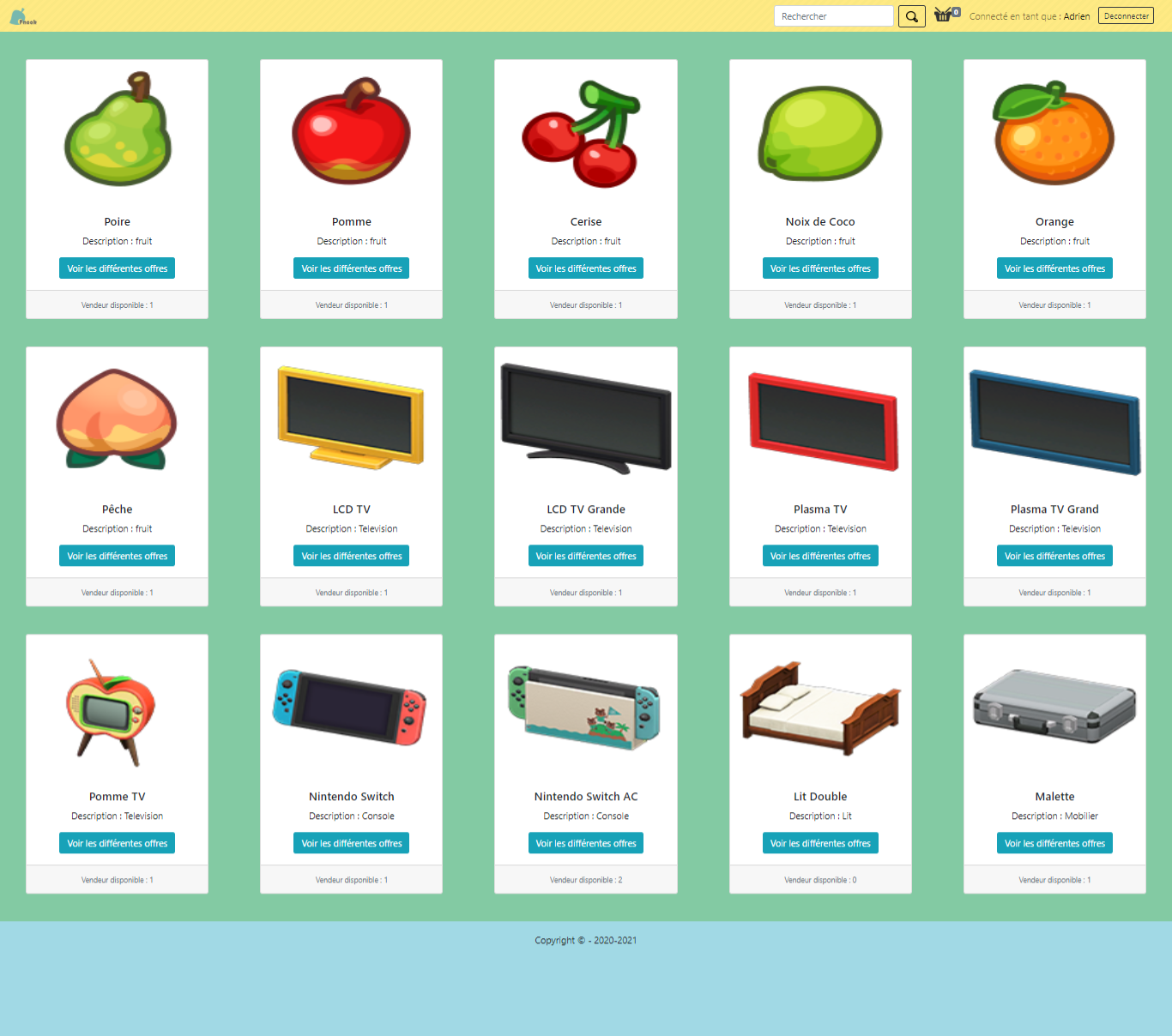
* « url\_api » /mcd
  + GET
  + Retourne au format JSON le modèle de donnée du projet
  + Dictionnaire
  + Droit : Tous

1. **Depuis l’IHM**

url\_ihm: <http://localhost:3030/>

Le front est assez simple à utiliser, il fonctionne aux cliques. L’authentification n’est pas nécessaire à la consultation. Néanmoins il vous faudra vous connecter pour accéder à l’ensemble des options.

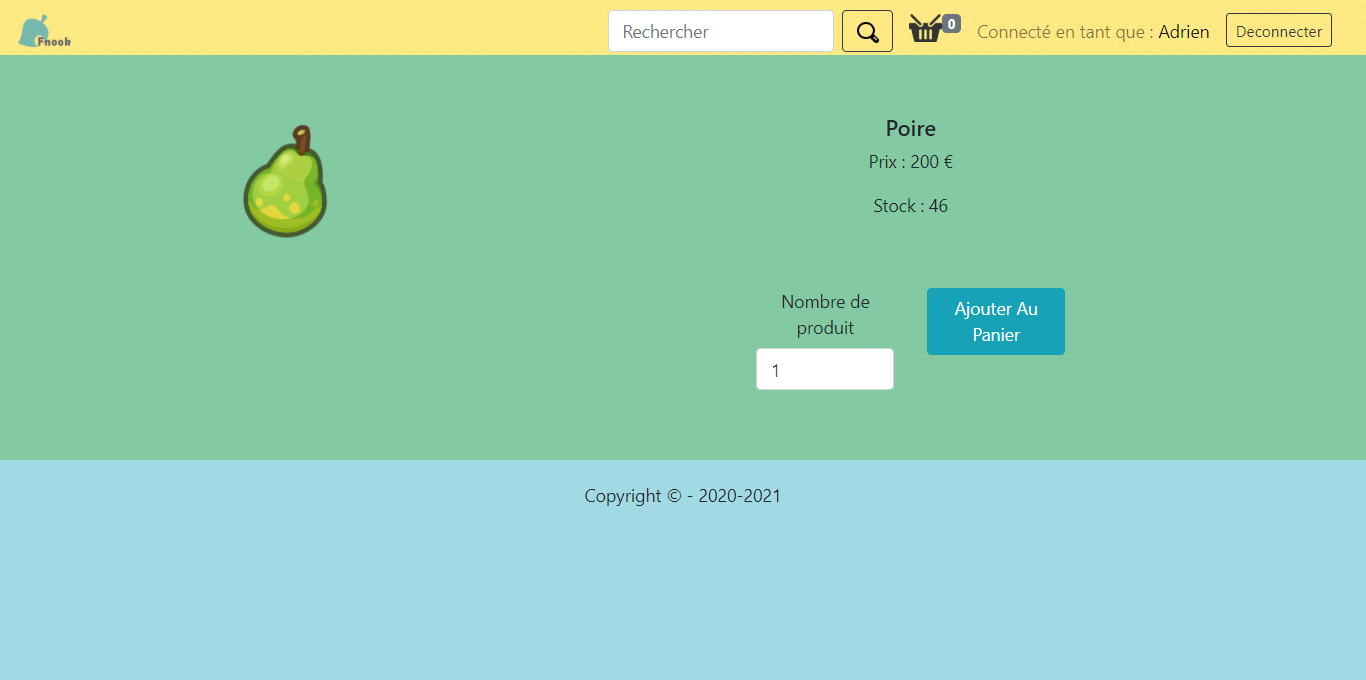
* <url\_ihm> /
  + Page d’accueil
  + Elle affiche l’ensemble des produits mères.



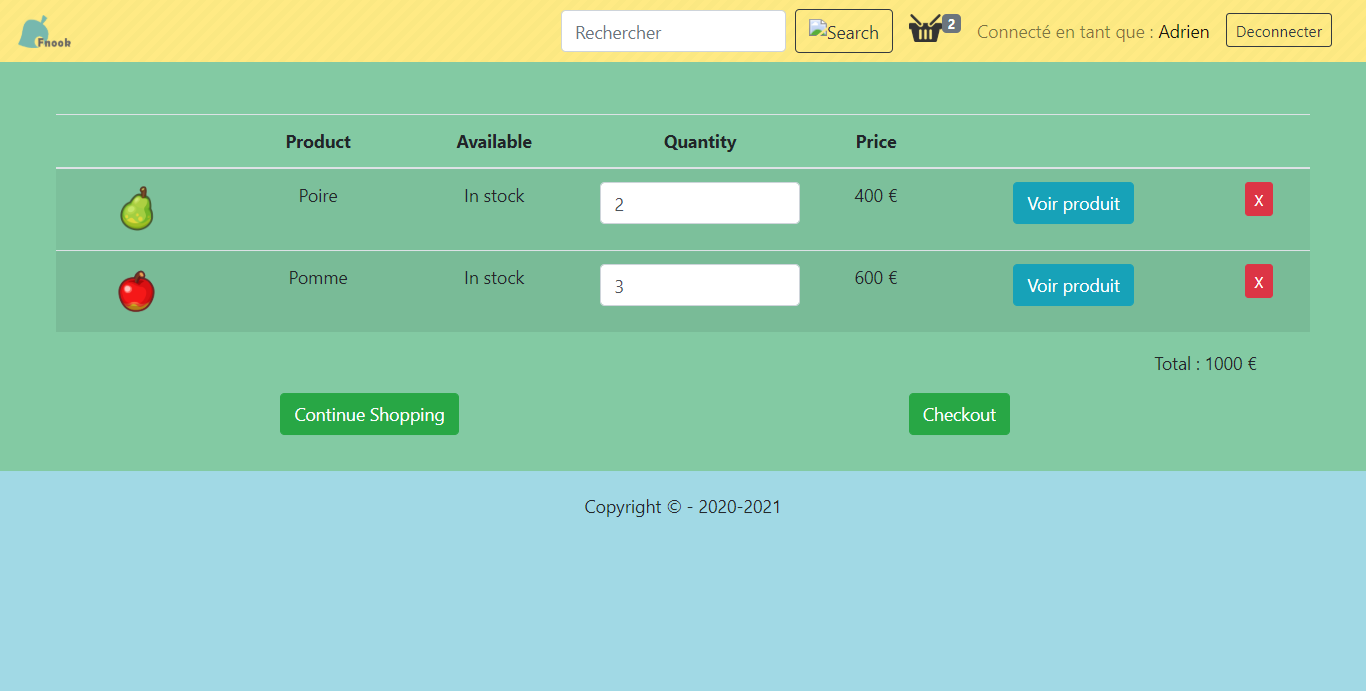
* <url\_ihm> /product/ :idmotherproduct
  + Affiche l’ensemble des produits lié à un produit mère



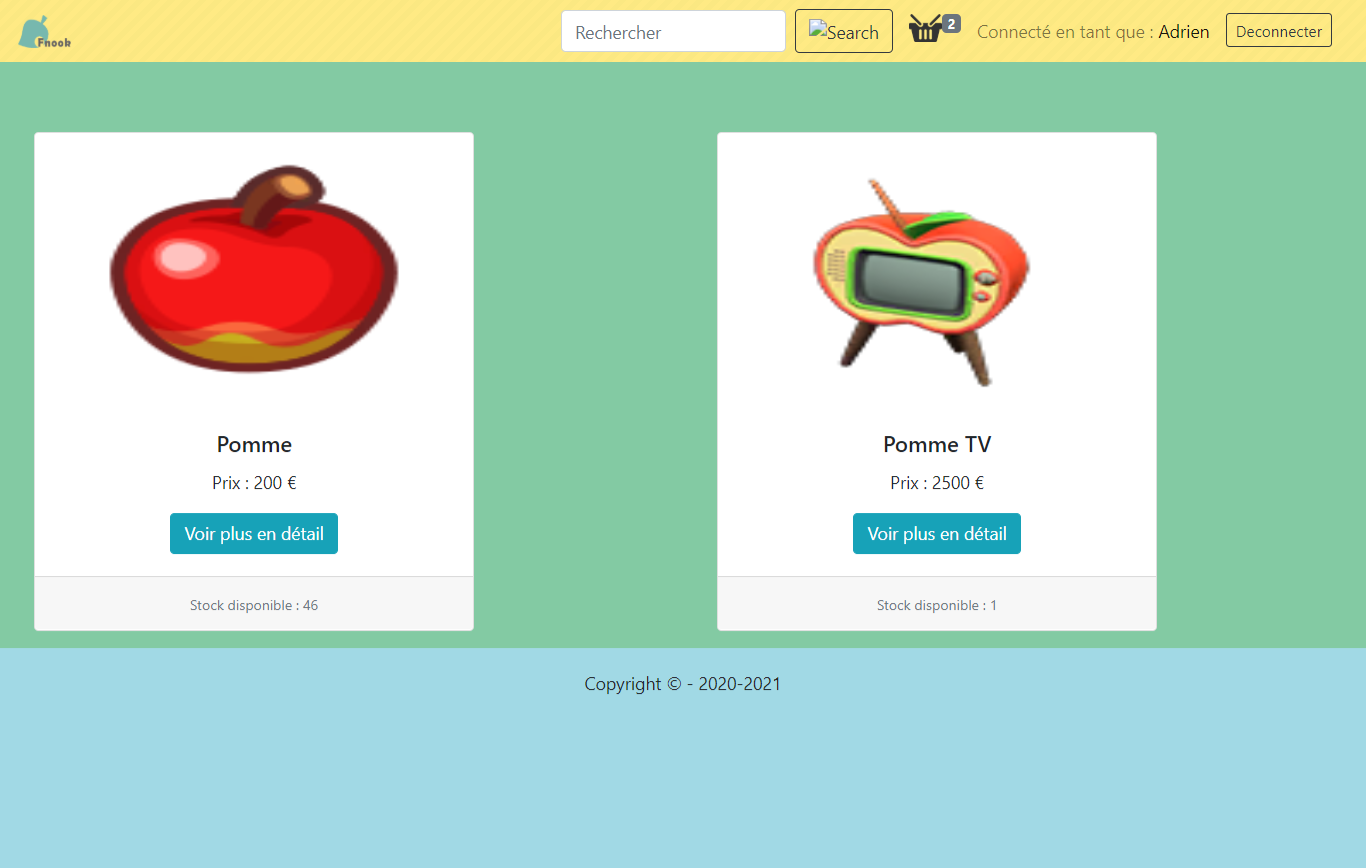
* <url\_ihm>/productDetail/ :idproduct
  + Affiche en détail un produit à partir de l’identifiant saisi
  + Possibilité de l’ajouter au panier



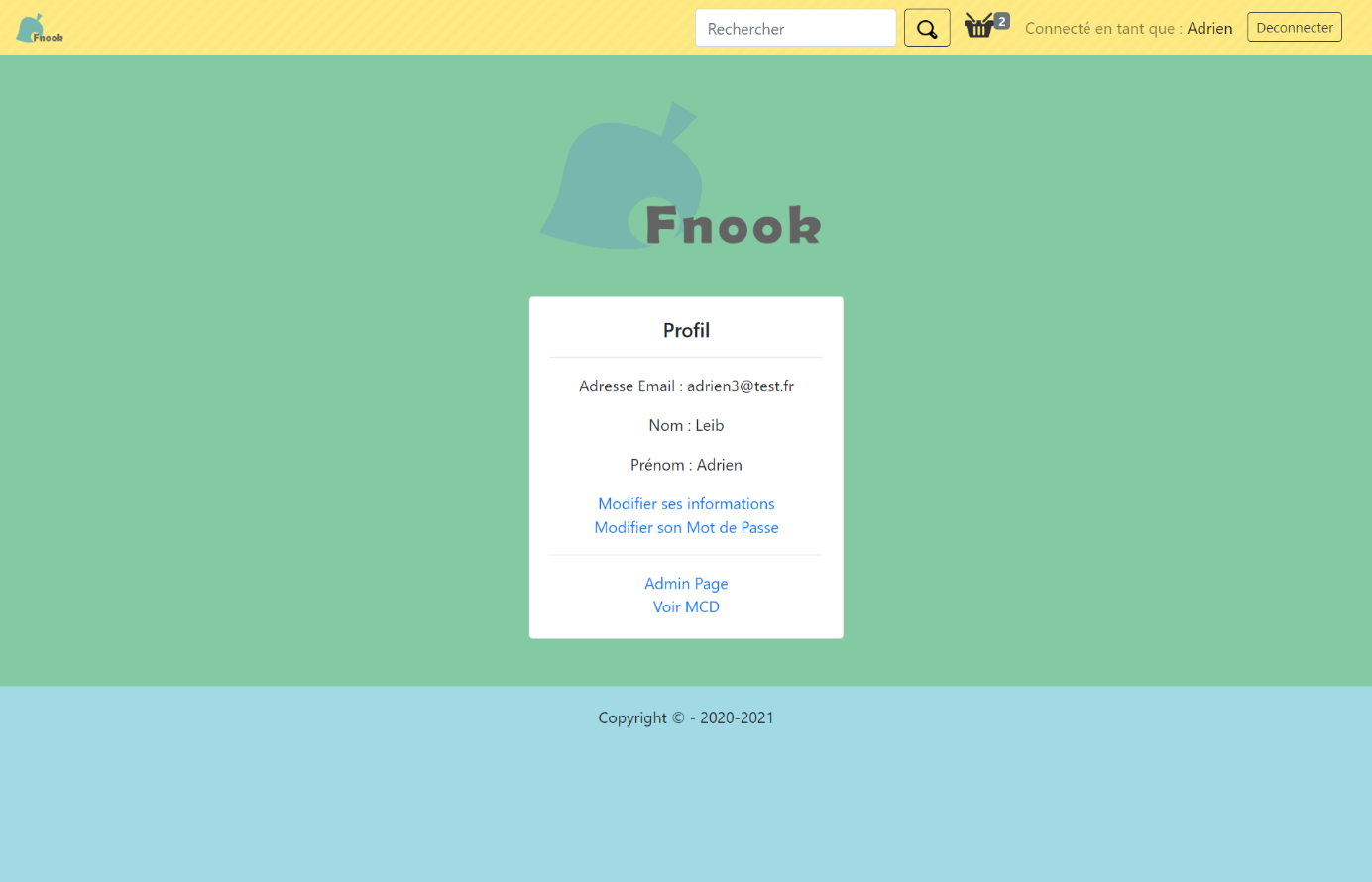
* <url\_ihm>/shopCart
  + Affiche l’ensemble des produits ajouté au panier
  + Si vous appuyé sur checkout en étant connecté cela créera une facture et supprimera le panier.



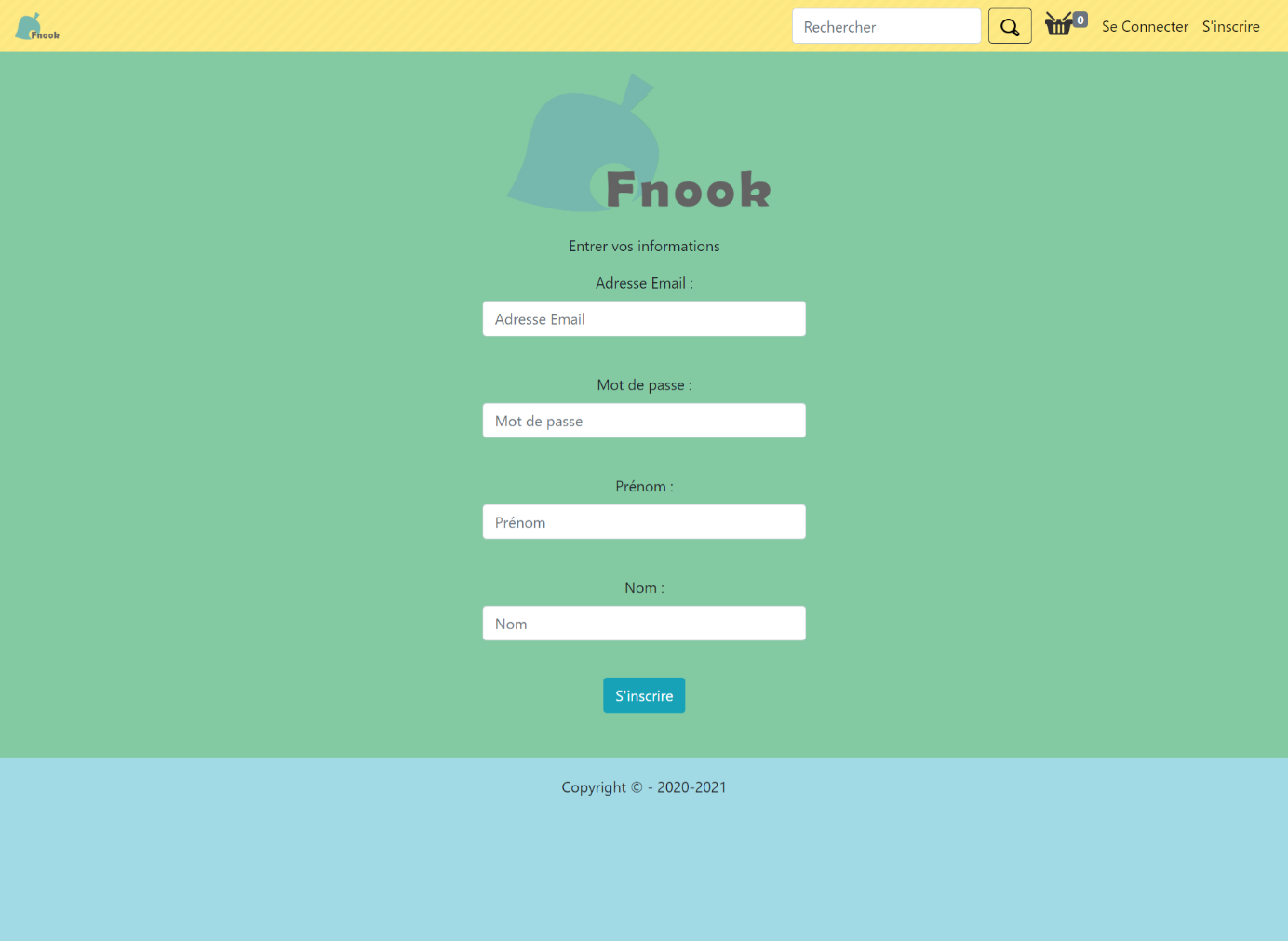
* <url\_ihm>/search/ :name
  + Affiche l’ensemble des produits dont le nom contient la valeur « :name »



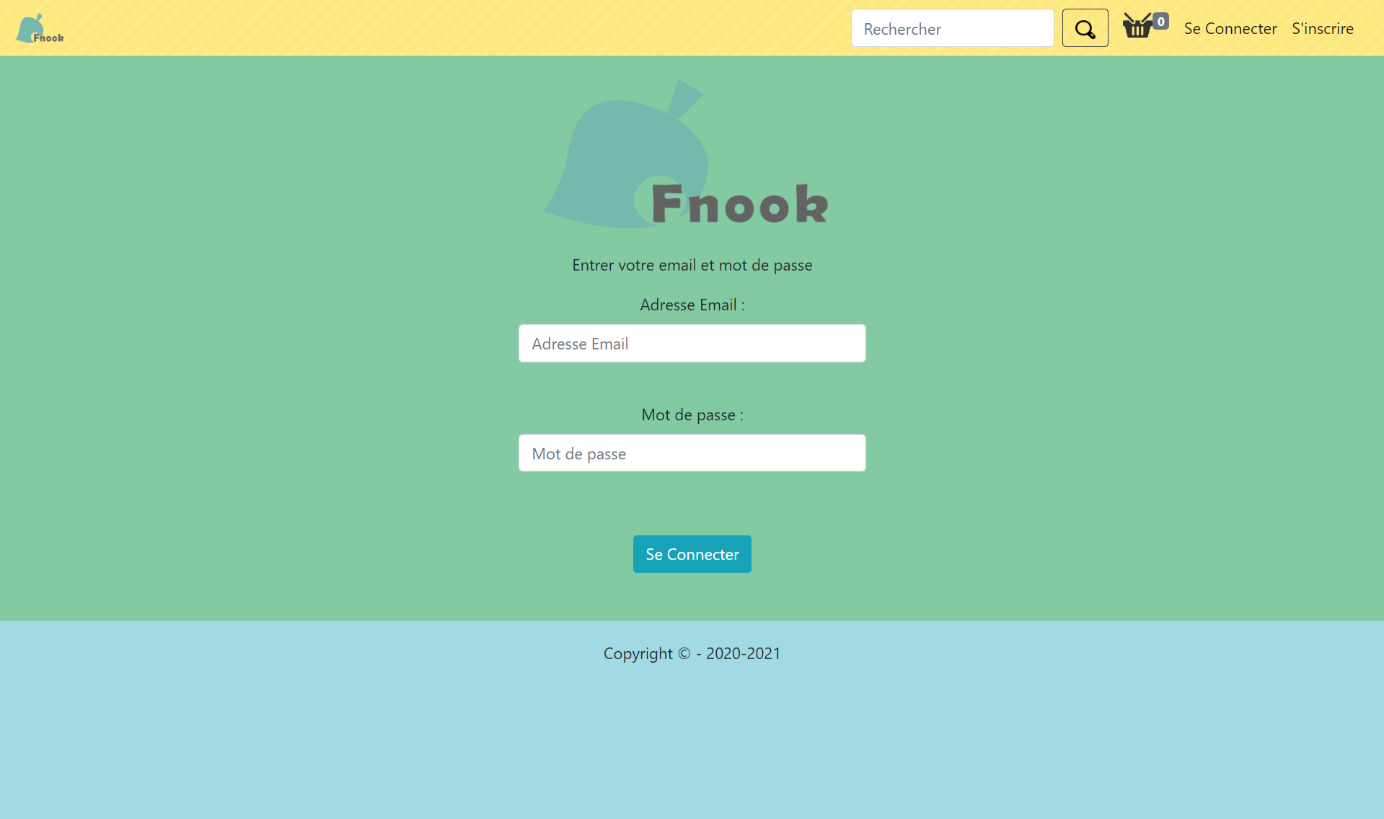
* « url\_ihm »/account
  + Affiche l’ensemble des données du profil
  + Est accessible uniquement si visiteur est connecté.



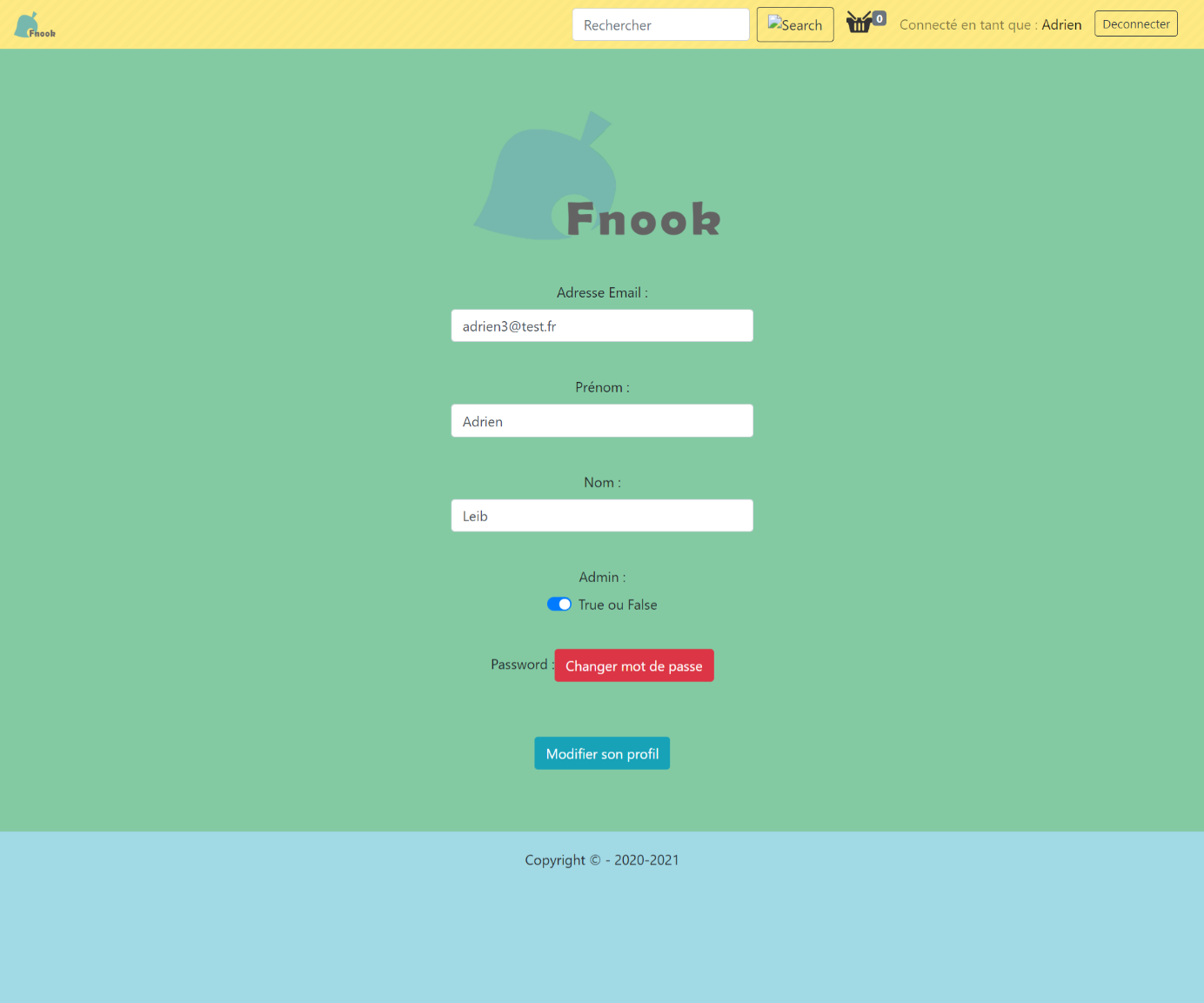
* « url\_ihm »/signup
  + Formulaire d’inscription



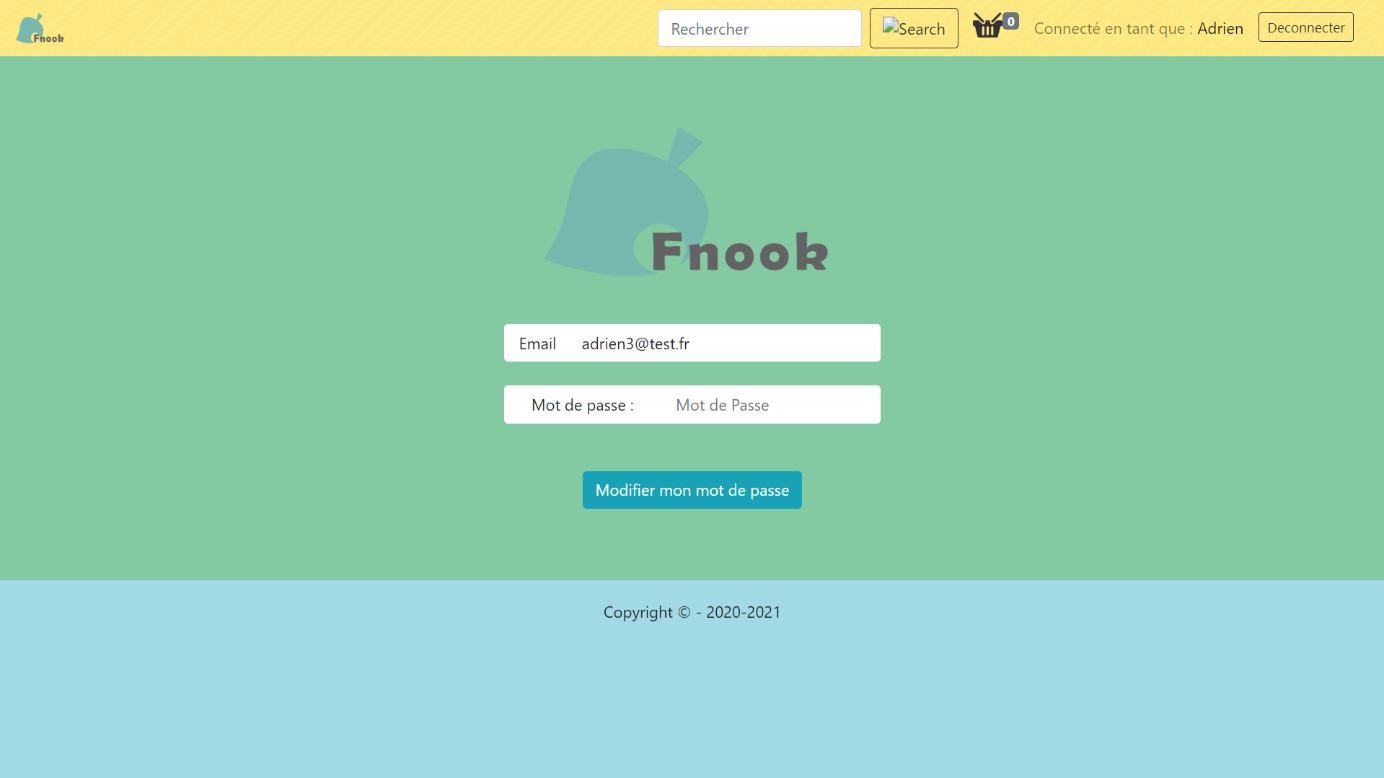
* « url\_ihm »/login
  + Formulaire de connexion



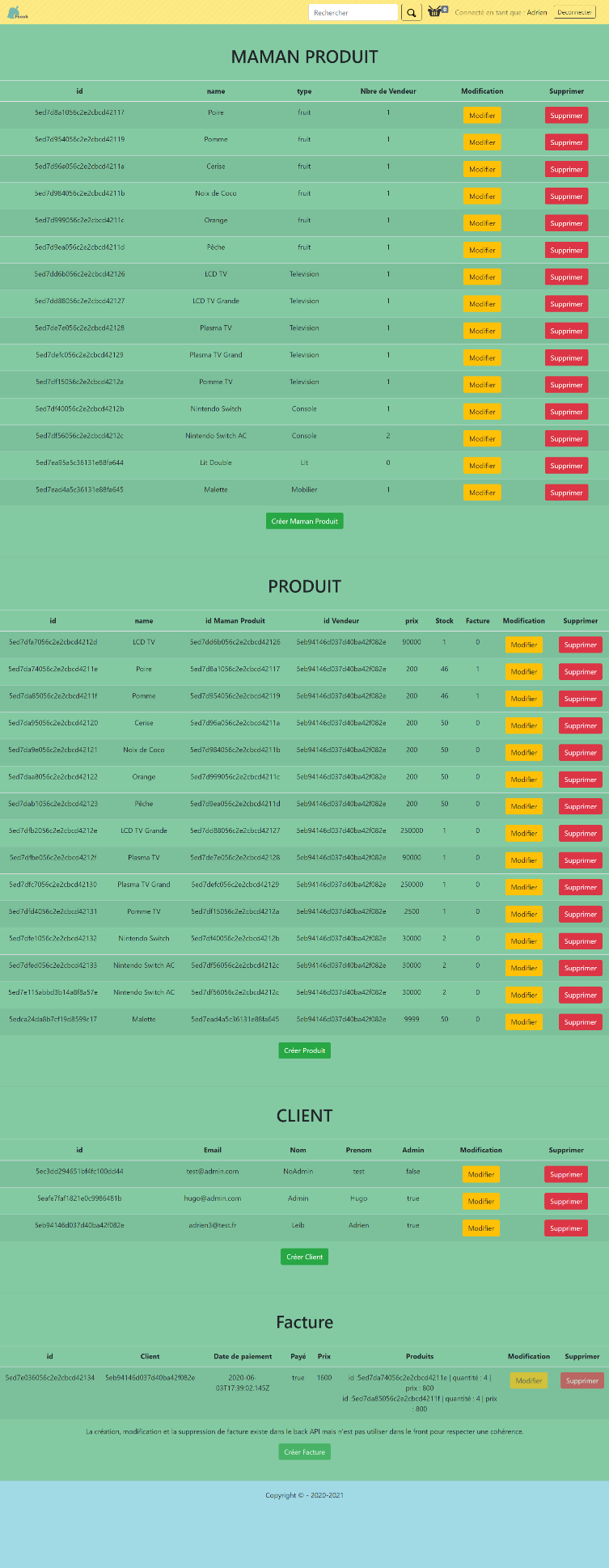
* « url\_ihm »/ClientUpdate/:id
  + Formulaire qui permet de modifier un utitilisateur.



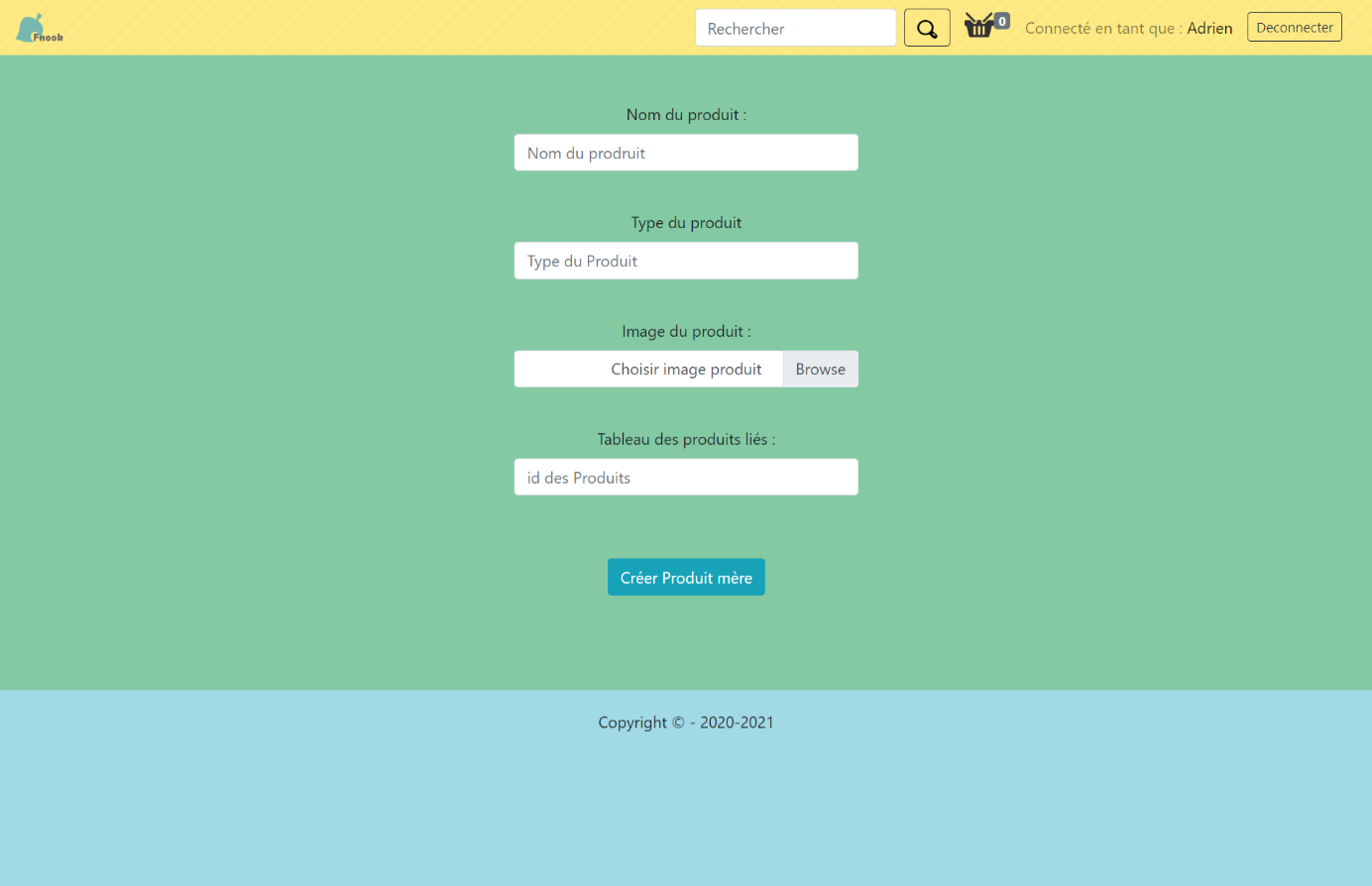
* « url\_ihm »/ClientPasswordUpdate/ :id
  + Formulaire qui permet de modifier un mot de passe



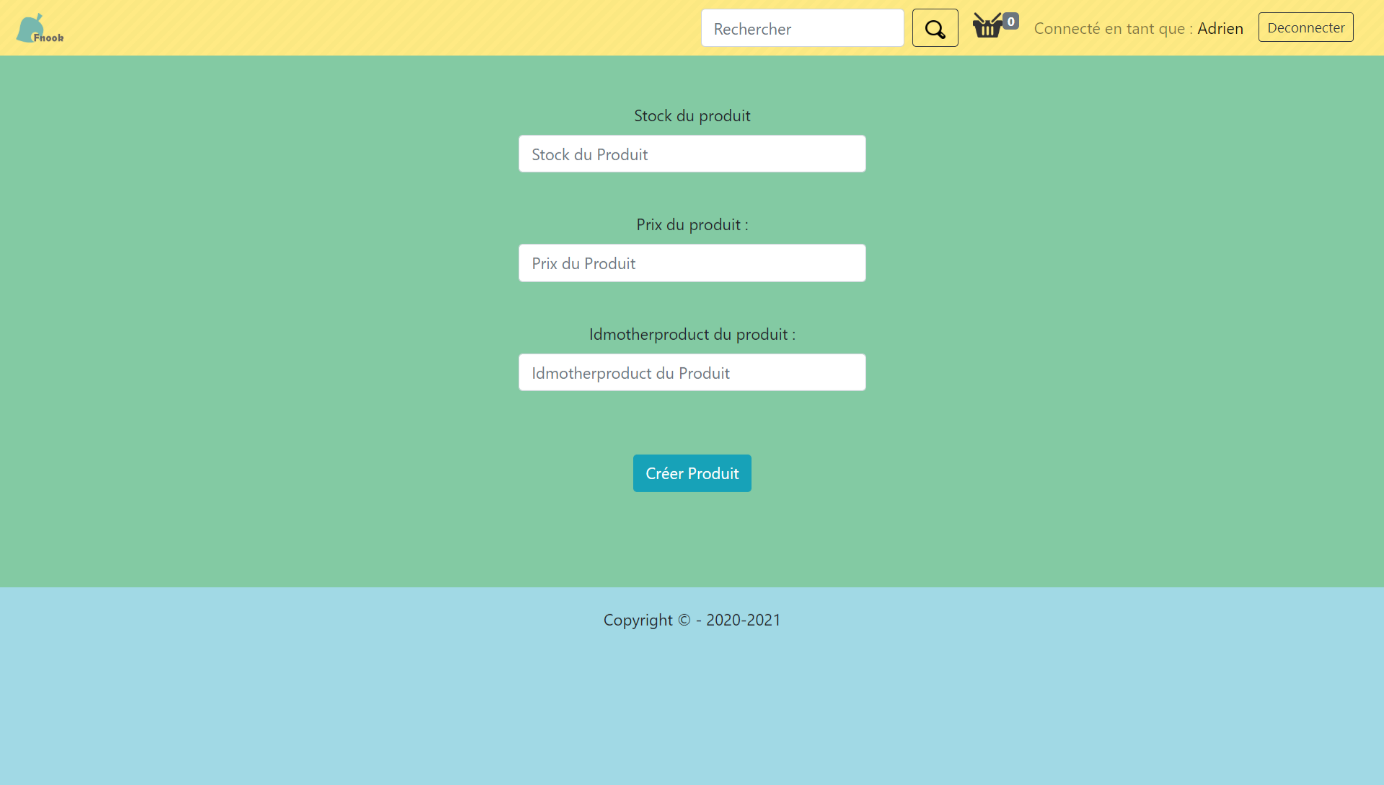
* « url\_ihm »/AdminPage
  + Page Web qui regroupe l’ensemble des données de l’api qu’un administrateur peut consulter.
  + L’administrateur a également la possibilité de modifier, supprimer chaque élément.



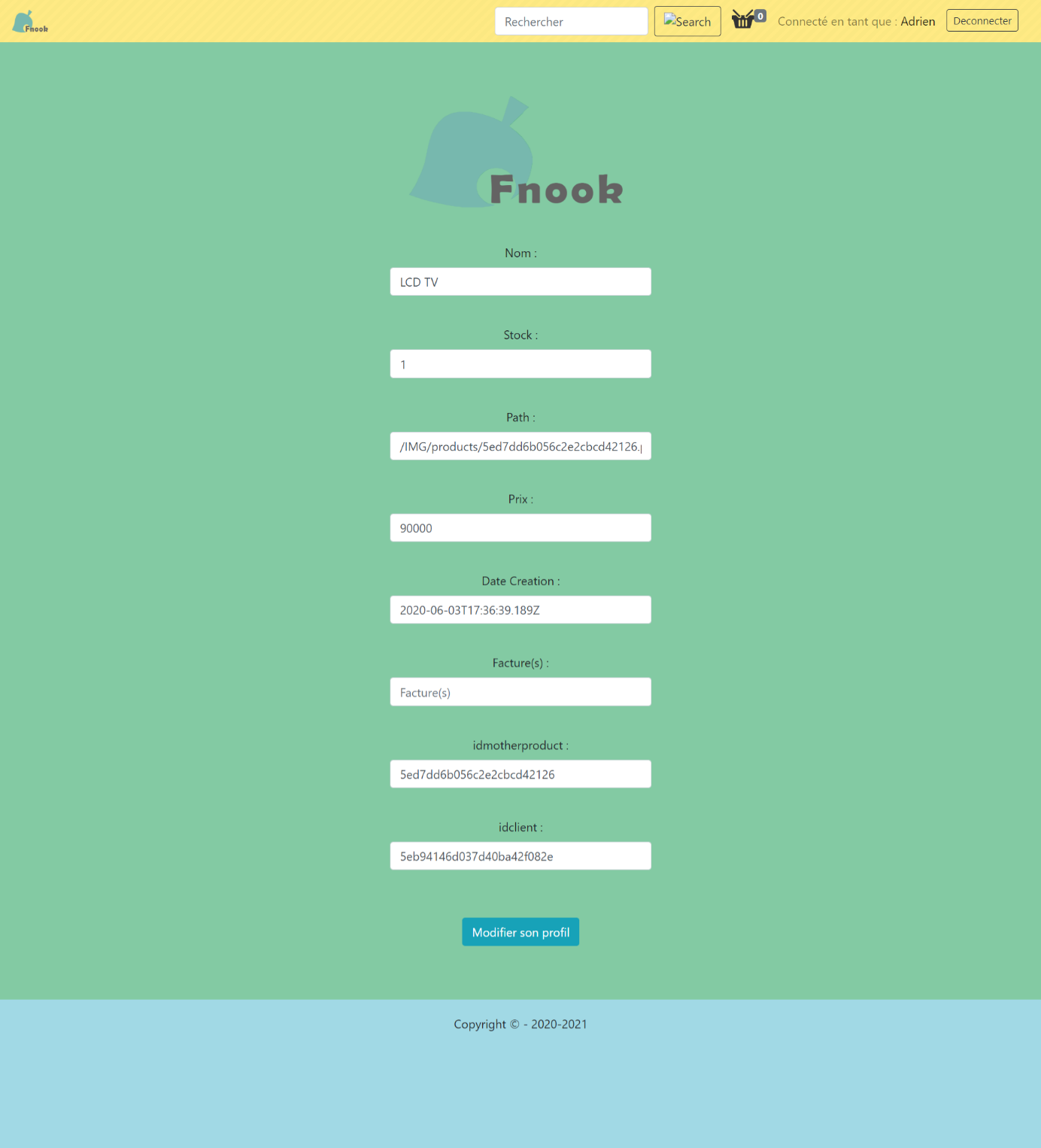
* « url\_ihm » /createMotherProduit
  + Formulaire pour créer un produit mere
  + Accessible uniquement pour les administrateurs



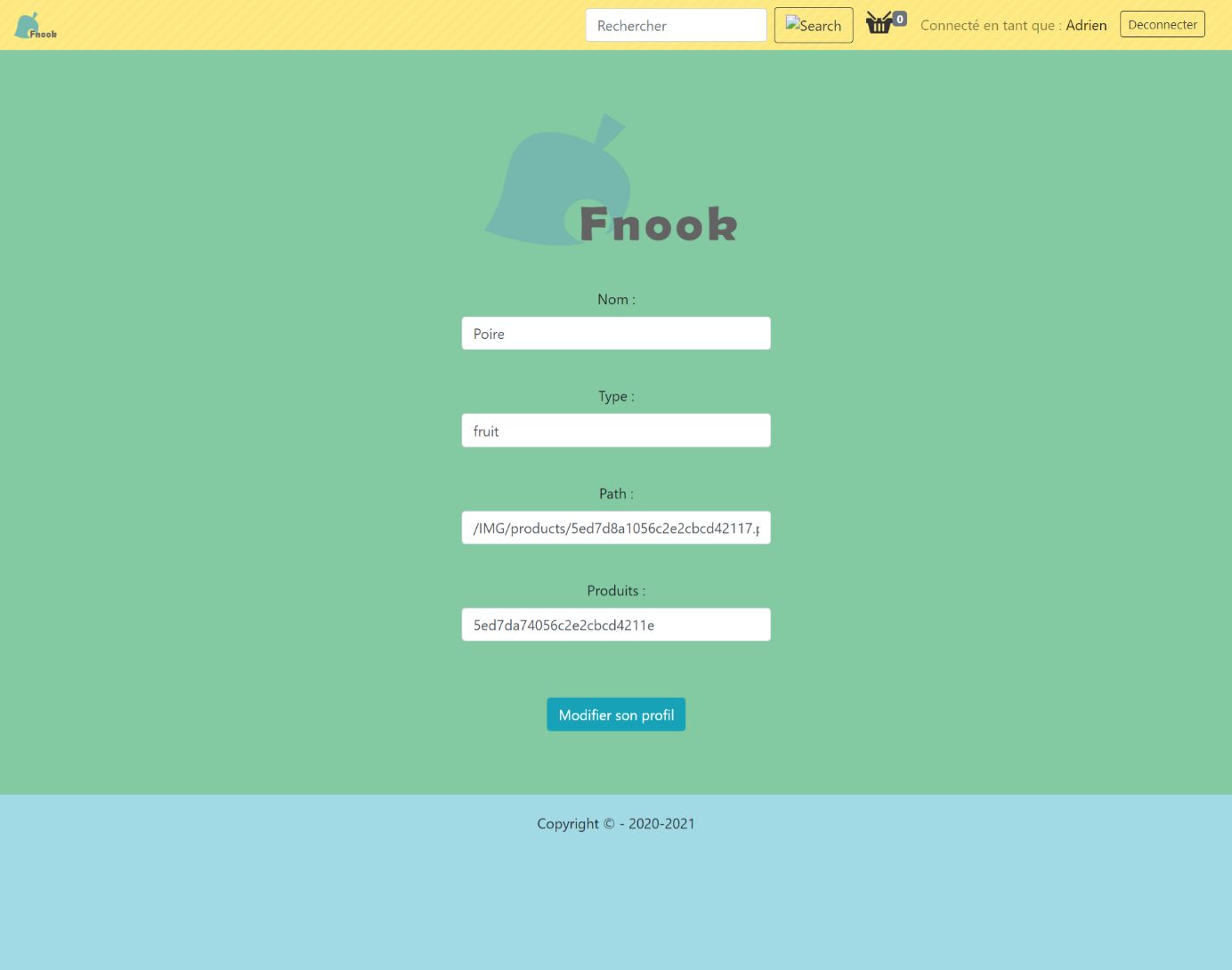
* « url\_ihm » /createProduit
  + Formulaire pour créer un produit lié à un produit mere
  + Accessible aux utilisateurs connectés



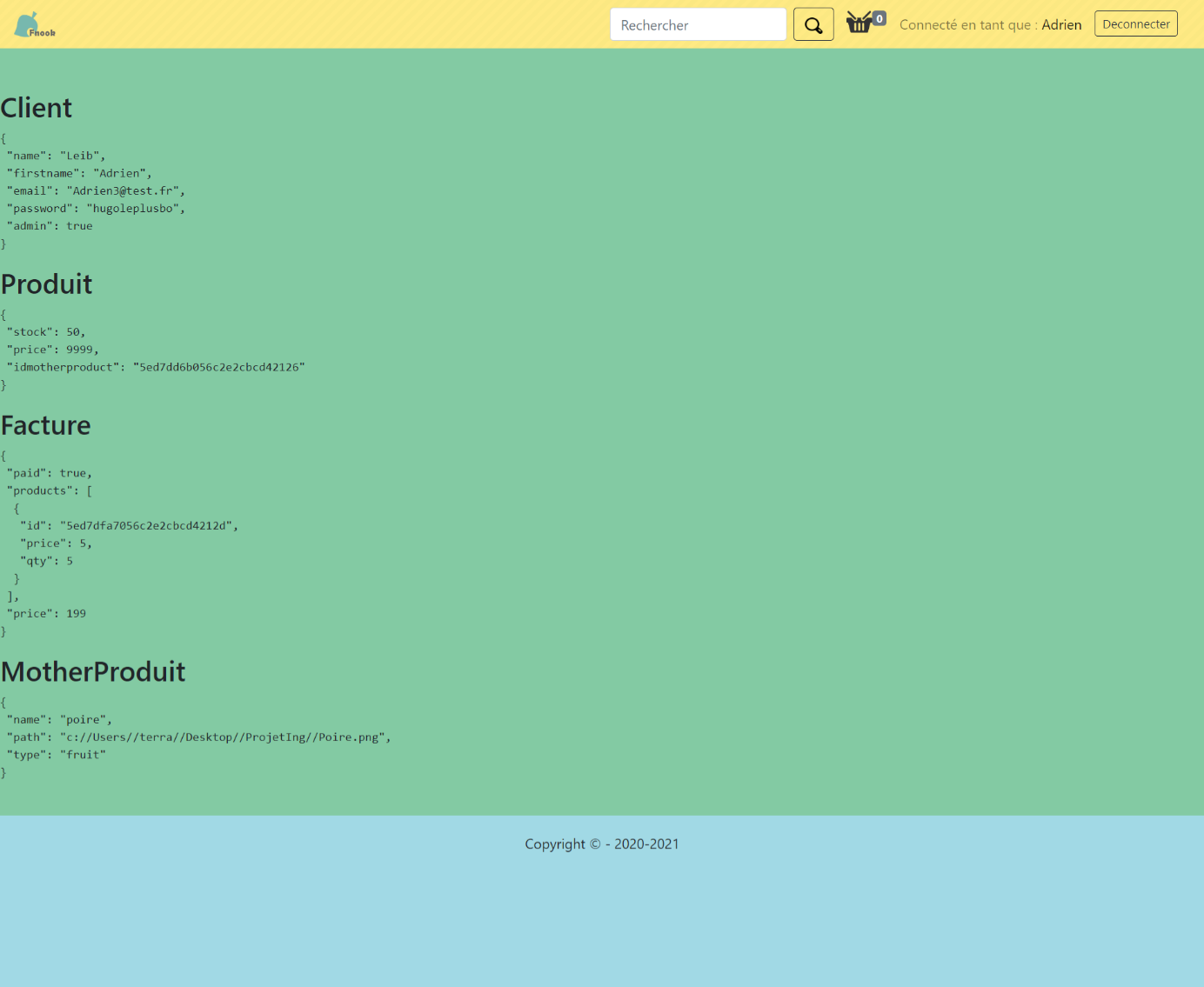
* « url\_ihm »/productUpdate/ :id
  + Formulaire pour Modifier un produit
  + Accessible à l’utilisateur vendeur et à un administrateur



* « url\_ihm »/motherproductUpdate/ :id
  + Formulaire pour modifier un produit mere à partir de son identifiant
  + Accessible à un administrateur



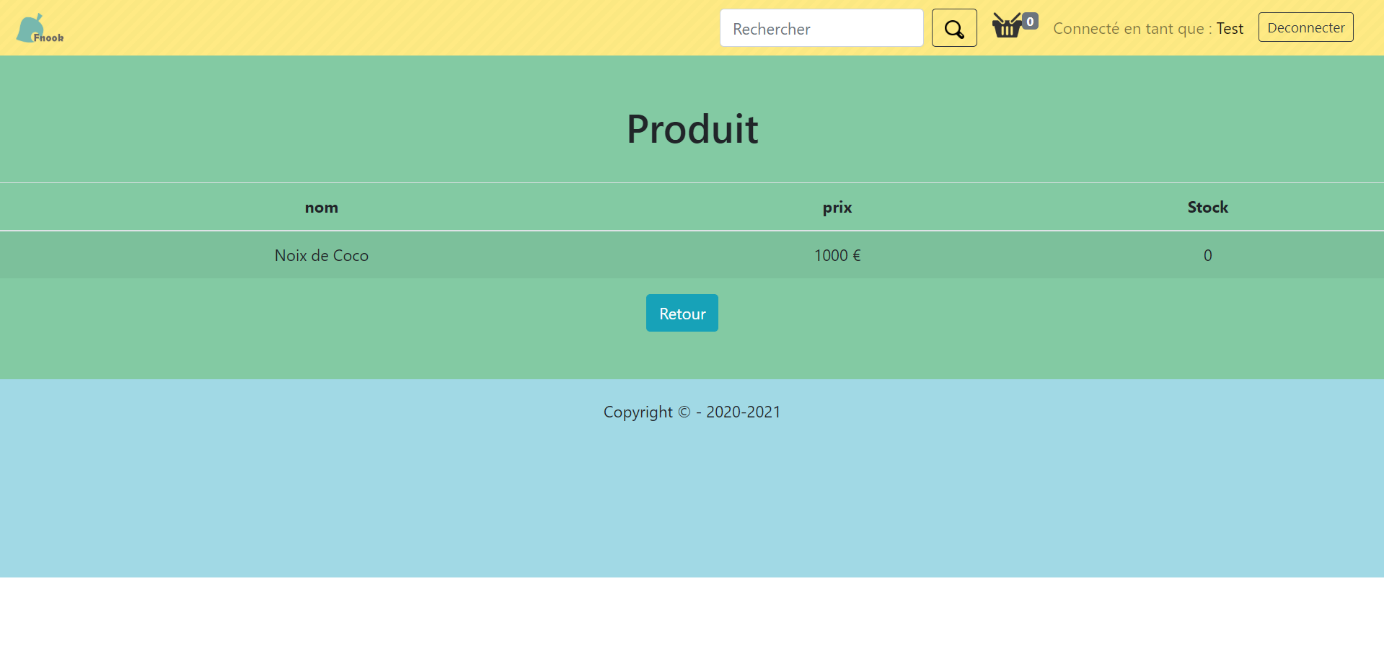
* « url\_ihm » /model
  + Affiche les modèles.



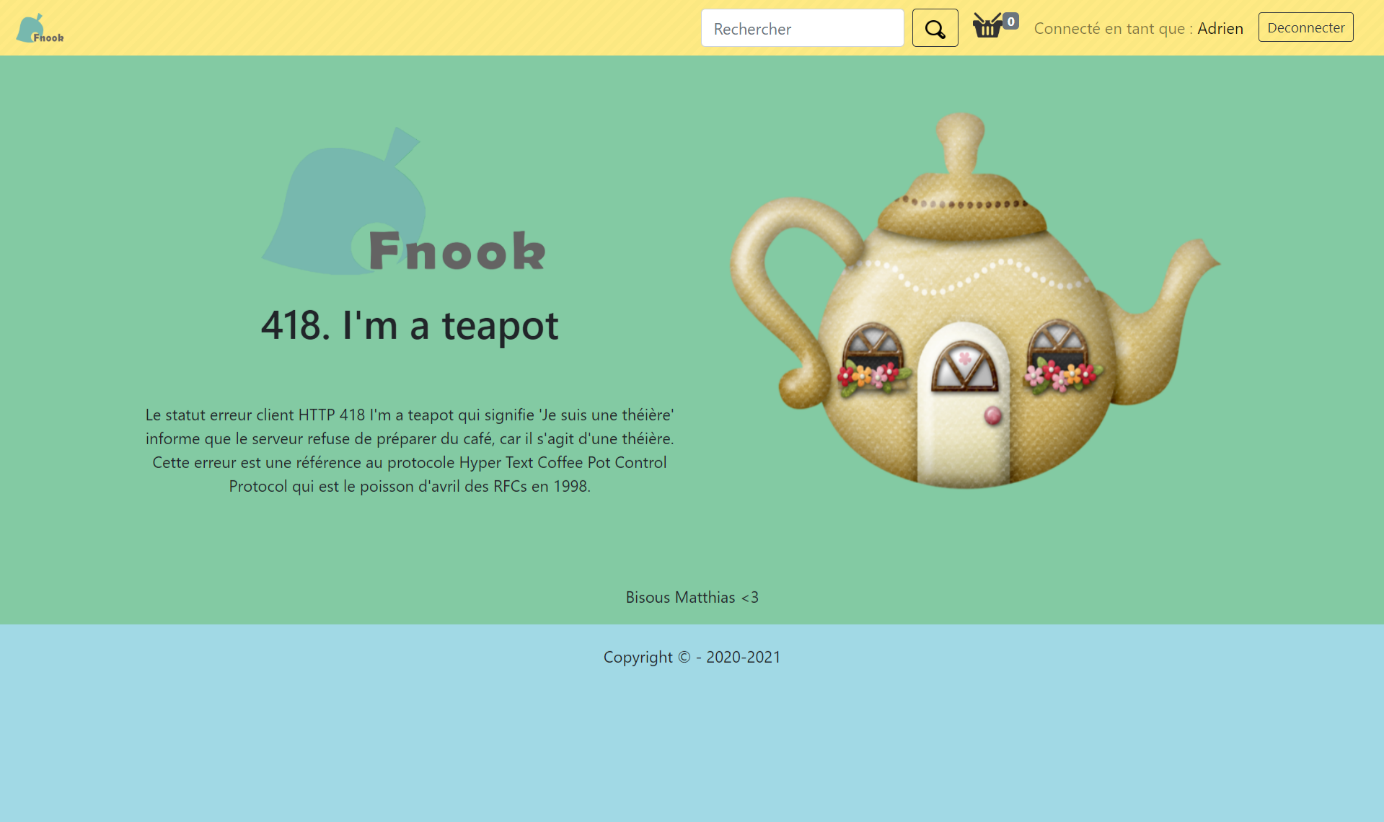
* « url\_ihm »/invoice/ :id
  + Affiche les factures liées à un user



* « url\_ihm »/voirProduitClient/ :id
  + Affiche les produits vendus par un client



* « url\_ihm » /faitMoiUnCafe
  + Erreur 418



* « url\_ihm »/404
  + Erreur 404

